

What is claimed is:

1. A method comprising
determining an address for a burst write of a command and arguments of the command based upon an argument count of the command, and
in response to performing the burst write, writing the arguments and the command starting from the address determined based upon the argument count.
2. The method of claim 1 wherein writing the arguments and the command comprises writing the arguments prior to writing the command.
3. The method of claim 1 wherein writing the arguments and the command comprises writing the arguments in a reverse order that results in a last argument of the command being written prior to a first argument of the command being written.
4. The method of claim 3 wherein writing the arguments and the command further comprises writing the command after writing the arguments.
5. The method of claim 1 further comprising performing a burst read of status and any outputs of the command.
6. The method of claim 5 wherein performing the burst read comprising reading the status prior to reading the any outputs.
7. The method of claim 5 further comprising determining an output count that is indicative of a number of outputs to be generated as a result of the command, wherein performing the burst read ends based upon the output count.

8. A method comprising
receiving a plurality of arguments of a command during a burst write, and
after receiving the plurality of arguments, receiving the command during
the burst write.

9. The method of claim 8 wherein receiving comprises receiving the
plurality of arguments in a reverse order in which a last argument of the
command is received prior to receiving a first argument of the command.

10. The method of claim 8 further comprising storing the plurality of
arguments and the command in contiguously writable registers.

11. The method of claim 8 further comprising executing the command
based upon the plurality of arguments in response to receiving the command.

12. The method of claim 11 further comprising storing status and outputs
resulting from executing the command in contiguously readable registers.

13. The method of claim 11 further comprising
receiving a burst read for status and outputs of the command, and
transferring the status of the command during the burst read, and
after transferring the status, transferring the outputs of the command.

14. An apparatus comprising
a command register to store a command,
a plurality of argument registers to store arguments of the command,
logic to execute the command based upon the arguments, wherein
the plurality of argument registers and the command register are
contiguously writable and are arranged such that a burst write of a command and
its arguments results in the arguments of the command being stored in the
plurality of argument registers before the command is stored in the command
register.

15. The apparatus of claim 14 wherein the command register follows the
plurality of argument registers.

16. The apparatus of claim 15 wherein the plurality of argument registers
are arranged in a reverse order that results in a last argument of the command
being stored before a first argument of the command is stored.

17. The apparatus of claim 14 further comprising
a status register to store status of the command, and
a plurality of output registers to store any outputs of the command,
wherein

the status register and plurality of output registers are contiguously
readable.

18. The apparatus of claim 14 wherein the status register and the plurality
of output registers are arranged such that the status is transferred prior to the
any outputs of the command in response to a burst read.

19. A machine-readable medium comprising a plurality of instructions that, in response to being executed, result in a computing device

determining an address for a burst write of a command and arguments of the command based upon an argument count of the command, and

in response to performing the burst write, writing the arguments and the command starting from the address determined based upon the argument count.

20. The machine-readable medium of claim 19 wherein the plurality of instructions further result in the computing device writing the arguments prior to writing the command.

21. The machine-readable medium of claim 19 wherein the plurality of instructions further result in the computing device writing the arguments in a reverse order that results in a last argument of the command being written prior to a first argument of the command being written.

22. The machine-readable medium of claim 19 wherein the plurality of instructions further result in the computing device writing the command after writing the arguments.

23. A system comprising
a display codec comprising a plurality of argument registers and a command register arranged to support burst writes of commands having different argument counts, and

a graphics controller to burst write the commands and their arguments to the plurality of argument registers and command registers of the display codec.

24. The system of claim 23 wherein the graphics controller further provides the graphics controller with pixel data via a video interface bus that couples the graphics controller and the display codec, and performs the burst write via a control bus that couples the graphics controller and the display codec.

25. The system of claim 23 wherein the control bus comprises an I2C bus.

26. The system of claim 23 wherein the plurality of argument registers and the command register are contiguously writable and the command register follows the plurality of arguments.

27. The system of claim 23 wherein the graphics controller during the burst write writes any arguments of a command to the plurality of argument registers of the display codec, and after writing the any arguments of the command to the plurality of argument registers, writes the command to the command register of the display codec.

28. The system of claim 23 wherein the graphics controller during the burst write

writes a last argument of a command to the plurality of argument registers of the display codec before writing the first argument of the command to a first argument register of the plurality of argument registers, and

after writing any arguments of the command to the plurality of argument registers, writes the command to the command register of the display codec.

29. The system of claim 23 wherein the display codec executes a command based upon its stored arguments in response to the command being stored in the command register.